



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/675,969

10/02/2003

Hong-Ki Kim

123016-05004702

6294

22429

7590

09/22/2010

LOWE HAUPTMAN HAM & BERNER, LLP
1700 DIAGONAL ROAD
SUITE 300
ALEXANDRIA, VA 22314

EXAMINER

THERIAULT, STEVEN B

ART UNIT

PAPER NUMBER

2179

MAIL DATE

DELIVERY MODE

09/22/2010

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/675,969	Applicant(s) KIM ET AL.	
	Examiner STEVEN B. THERIAULT	Art Unit 2179	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 29 January 2010.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-23 and 37-54 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-23, 37-54 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

Art Unit: 2179

DETAILED ACTION

1. This action is responsive to the following communications: pre-appeal conference request filed 01/26/2010 and a decision mailed 07/12/2010.
2. Claims 1-23, 37-54 are pending in the case.

Applicant is advised that a new examiner has been assigned to the application.

Response to Arguments

3. Applicant's argument's, see page 4, filed 01/29/2010, with respect to the rejection(s) of claim(s) 1-23, 37-54 under Bogdan in view of Pham have been fully considered and are persuasive. Specifically, applicant argues that Bogdan in view of Pham do not backing up the properties occurs automatically in response to user input (see argument that the rejection lacks clarity page 3, bottom and page 4, top). However, it is noted that a new examiner has been assigned to the application and a new ground of rejection is presented below based on a new search and consideration of prior art based on claims as of 06/11/2009, and the previous rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection (below) is made over Wen, in view of Straub, in further view of Axialis.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966),

that are applied for establishing a background for determining obviousness under 35

U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.

Art Unit: 2179

3. Resolving the level of ordinary skill in the pertinent art.
 4. Considering objective evidence present in the application indicating obviousness or nonobviousness.
5. **Claim 1-23, 37-42, and 53-54 are rejected under 35 U.S.C 103(a) as being unpatentable over Wen et al. (hereinafter Wen) U.S. Patent No. 6590590 issued July 8, 2003, in view of Straub et al. (hereinafter Straub) U.S. Patent No. 6091411 issued July 18, 2000, in further view of Axialis IconWorkshop et al (hereinafter Axialis).**

In regard to **Independent claim 1**, Wen teaches a method of controlling an icon appearance of a display system having a display screen, the method comprising:

- Displaying an icon control window on the display screen, the icon control window including at least one sample icon for a user's preview (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61). Wen teaches a control window that allows the user to see an icon and a preview of the sampled icon in the preview window.
- Changing the at least one sample icon's appearance according to inputs for a new icon appearance being received from a user through the icon control window(See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-21 and column 4, lines 33-61). Wen teaches the user can change the appearance of the icon by cropping it and the cropped image of the icon will be shown in the preview window. The user manipulates the cropping window to change the image.
- Backing up display properties of the display system which are currently set for an original icon appearance by generating a first registry sub key in a memory of the display system and storing the display properties in a corresponding registry (See column 4, lines 33-61 and column 6, lines 8-12) Wen teaches the user saves the file path of the old icon while changing the icon

Art Unit: 2179

appearance. Wen states the user renames the old file name of the old icon. The user then saves the target image file of the new icon as the default icon name, thus changing the system icon.

- Changing the icon appearance of the display system by changing the display properties in accordance with the user inputs, through the icon control window and is performed immediately prior to changing the at least one sample icon's appearance(See column 4, lines 33-61 and column 6, lines 8-12 and figure 7). Wen shows in figure 7 that the update to change the icon appearance is reflected in the control window and the appearance is changed when the user crops or adjusts the image. The update to the new icon is performed prior to viewing the sample icon in the preview window.

Wen does not expressly teach

- wherein backing up the display properties occurs automatically in response to the inputs for a new icon appearance being received from the user
- if the display properties are determined to be valid, changing the display properties of the icon

Straub teaches a dynamic user interface update system that allows the user to make changes to icons, display colors, backgrounds, etc and in response to the update a service is activated to backup the display properties in the registry 66, when a smart folder update is specified (see column 8, lines 25-45). The web update service is set when the user changes the interface display properties from a window or dialog box (See column 8, top). Straub teaches the web update service "can" perform updates by unattended scheduled downloading and "can also" perform updates by background downloading when the system is connected to the Internet and performs the update "automatically" according to the registry entries (see col. 10, lines 1-31). Straub also verifies the "validity" of the WCF (web collection file) using digital signatures where the WCF points to the new interface theme icons. Thus Straub "suggests" automatically updating an icon, where the user makes adjustments to the displayed theme and the registry entry and web update are updated. The system verifies the validity of the theme change and icon and automatically backs up the display properties via the web update service. Straub and Wen teach

Art Unit: 2179

updating a user desktop appearance, via dialog boxes open on the interface. They both teach updating registry files. They both teach allowing the user to specify how the appearance of the icon is set. The difference between Straub and Wen is in Wen the user specifies the path in the registry to store the old icon and in Straub the system automatically updates a folder that refers to the registry entries when the user makes an update. Nonetheless, the outcome on the interface for both will result in a change to an icon appearance by changing the registry values used by the interface to point to the new image location when rendering the interface.

Accordingly, it would have been obvious to the skilled artisan at the time of the invention having the teachings of Straub and Wen in front of them to modify the manual process of updating registry values with the automatic system of Straub for the purposes of updating the interface without user input and to keep track of changes to interface appearances (see column 6, lines 55-67, column 7, lines 1-15, and 50-67, column 8, top, column 10, lines 1-31 and column 14, lines 25-67 and column 15, lines 1-45). The motivation to combine Straub with Wen comes from the suggestion in Straub that the windows interface provides for many features that allow the user to perform interface customizations to icons and other interface elements (See column 2, lines 12-55 and col. 6, lines 55-67) and to store and automatically update the interface elements on the computer (See col. 7, lines 1-50).

Even though Wen teaches a preview window that displays a preview icon of the user's manipulated icon in the crop interface and the crop icon can be considered a sample icon, the text does not expressly recite that the crop window 21 displays the sample icon. The text recites displaying the old icon, while allowing the user to adjust the old icon displaying the preview icon. The user can nonetheless make multiple crop changes to window 21 without actually changing the system icon appearance. The window 21 is considered as displaying at least one sample icon that is visible to the user. The user makes a change and a preview of the icon is displayed, adjacent to the adjusted icon. The claim does not specify user interface actions that may lead up to or prior to the displaying of the icon window, or that the user can or cannot perform additional steps to retrieve the previews. However, Wen does not state displaying preview icons in the plurality in the icon cropping manager window shown in figure 2.

Art Unit: 2179

Therefore, Wen in view of Straub does specifically recite

A sample icon for user's preview

Axialis teaches a Icon Workshop interface editing system that allows the user to see multiple preview or sample icons while making icon appearance changes (See Para 2, 3, 13-22, specifically page 21, that allows the user to see several sample icons while changing the size of the icon). Axialis also teaches an editor window view and it's built in explorer to browse icons and see the icons in a preview thumbnail (See page 50, middle top). Axialis teaches a editor that allows the user to make specific display appearance changes to an icon while displaying a preview or sample icon (See page 54) and a process to changes system icons (See page 57 and 114 - 116) by using a customize windows icon dialog box similar to Wen. Straub, Wen and Axialis are analogous art because they all teach a process of changing a displayed icons appearance. They all perform the function via a dialog with the user and they all allow the user to alter the appearance of the interface icon.

Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the invention having the teachings of Straub, Wen and Axialis in front of them to modify the system of Wen to show sample icons to the user comprising icons of different display properties such as size and color, etc. The motivation to combine Axialis with Straub and Wen comes from within Axialis to create different size icons so that the icons will be compatible with different screen configurations and so that the icon image formats will be valid with the current windows icon formats (See page 11 and 12). Moreover for the expressed feature of allowing the user to customize their windows icons (See page 2).

With respect to **dependent claims 2**, Wen teaches the method wherein the received inputs include at least one of an icon size, a vertical icon spacing, a horizontal icon spacing, an icon font size, and an icon font type (See Figure 2, the user can crop the image which effects its size, See also icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61)..

Art Unit: 2179

With respect to **dependent claims 3-7**, as indicated in the above discussion, Wen in view of Straub in further view of Axialis teaches each element of claim 1.

Wen and Straub do not expressly teach the method wherein the icon control window comprises:

- an icon size controller providing a plurality of selectable icon sizes for the user to select a desired icon size from the selectable icon sizes;
- a preview region including the at least one sample icon, the sample icon being resized when the desired icon size is selected through the icon size controller;
- an execution controller interfacing with the display system in order to change an icon size of the display system according to the selected icon size.
- wherein the icon size controller comprises a sliding bar with minimum and maximum icon sizes, the user selecting the desired icon size by moving a size indicator within the sliding bar.
- wherein the minimum and maximum icon sizes of the sliding bar are selected from a size range supported by the display system.
- wherein the icon size controller comprises a plurality of selectable buttons representing the plurality of selectable icon sizes, the user selecting the desired icon size by selecting one of the selectable buttons.
- wherein the plurality of selectable buttons include toggle buttons.

Axialis teaches a Icon Workshop interface editing system that allows the user to see multiple preview or sample icons while making icon appearance changes (See Para 2, 3, 13-22, specifically page 21, that allows the user to see several sample icons while changing the size of the icon). Axialis also teaches an editor window view and it's built in explorer to browse icons and see the icons in a preview thumbnail (See page 50, middle top). Axialis shows a plurality of size of the icons displayed in the interface and a sliding bar that allows the user to scroll through the icon sizes. Axialis teaches a table that is shown to the user to indicate the max. and min icon size for the given operating system. The user can browse the icon images and select or unselect the sample icons they wish to see within the interface. Axialis teaches a editor that allows the user to make specific display appearance changes to an icon while displaying a preview or sample icon (See page 54) and a process to changes system icons (See page 57 and 114 - 116) by using a customize windows icon dialog box similar to Wen. Straub, Wen and Axialis are analogous art because they all teach a process of changing a displayed icons appearance. They all

Art Unit: 2179

perform the function via a dialog with the user and they all allow the user to alter the appearance of the interface icon.

Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the invention having the teachings of Straub, Wen and Axialis in front of them to modify the system of Wen to show sample icons to the user comprising icons of different display properties such as size and color, etc. The motivation to combine Axialis with Straub and Wen comes from within Axialis to create different size icons so that the icons will be compatible with different screen configurations and so that the icon image formats will be valid with the current windows icon formats (See page 11 and 12). Moreover for the expressed feature of allowing the user to customize their windows icons (See page 2).

With respect to **dependent claim 8**, Wen teaches the method wherein the icon control window comprises:

a plurality of manual input controllers manually receiving the inputs from the user; a preview region including the at least one sample icon, the sample icon's appearance being changed according to the manually received inputs; and an execution controller interfacing with the display system for changing the display properties in accordance with the received user inputs (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61). Wen shows a plurality of input selectors around the cop window that allow the user to change the appearance of the icon by selecting portions of it. The selected portion is shown in the preview window.

With respect to **dependent claim 9**, Wen teaches the method wherein the user inputs comprises at least one of an icon size, a vertical icon spacing, a horizontal spacing, an icon font size, and an icon font type (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61). Wen teaches changing the size of the icon by cropping it.

With respect to **dependent claims 10-11**, as indicated in the above discussion, Wen in view of Straub in further view of Axialis teaches every element of claim 1.

Art Unit: 2179

Wen does not expressly recite where the display properties are determined to be valid based on a display properties table of the display system and displaying the icon control window on the display screen if the display properties are determined to be valid.

Straub teaches a dynamic user interface update system that allows the user to make changes to icons, display colors, backgrounds, etc and in response to the update a service is activated to backup the display properties in the registry 66, when a smart folder update is specified (see column 8, lines 25-45). Straub teaches a registry that stores display information in a database, which contains tables. The web update service is set when the user changes the interface display properties from a window or dialog box (See column 8, top). Straub teaches the web update service "can" perform updates by unattended scheduled downloading and "can also" perform updates by background downloading when the system is connected to the Internet and performs the update "automatically" according to the registry entries (see col. 10, lines 1-31). Straub also verifies the "validity" of the WCF (web collection file) using digital signatures where the WCF points to the new interface theme icons. Thus Straub "suggests" automatically updating an icon, where the user makes adjustments to the displayed theme and the registry entry and web update are updated. The system verifies the validity of the theme change and icon and automatically backs up the display properties via the web update service. Straub and Wen teach updating a user desktop appearance, via dialog boxes open on the interface. They both teach updating registry files. They both teach allowing the user to specify how the appearance of the icon is set. The difference between Straub and Wen is in Wen the user specifies the path in the registry to store the old icon and in Straub the system automatically updates a folder that refers to the registry entries when the user makes an update. Nonetheless, the outcome on the interface for both will result in a change to an icon appearance by changing the registry values used by the interface to point to the new image location when rendering the interface.

Accordingly, it would have been obvious to the skilled artisan at the time of the invention having the teachings of Straub and Wen in front of them to modify the manual process of updating registry values with the automatic system of Straub for the purposes of updating the interface without user input and to keep track of changes to interface appearances (see column 6, lines 55-67, column 7, lines 1-15,

Art Unit: 2179

and 50-67, column 8, top, column 10, lines 1-31 and column 14, lines 25-67 and column 15, lines 1-45).

The motivation to combine Straub with Wen comes from the suggestion in Straub that the windows interface provides for many features that allow the user to perform interface customizations to icons and other interface elements (See column 2, lines 12-55 and col. 6, lines 55-67) and to store and automatically update the interface elements on the computer (See col. 7, lines 1-50).

With respect to **dependent claim 12**, Wen teaches the method wherein the changing the at least one sample icon's appearance comprises:

determining whether the inputs for the new icon appearance are received through the icon control window; (See column 4, lines 30-61 and column 3, lines 1-22).

changing at least one of an icon size, vertical icon spacing, horizontal icon spacing, icon font size, and icon font type of the at least one sample icon according to the new icon appearance if the user inputs are received through the icon control window(See Figure 2, the user can crop the image which effects its size, See also icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61)..

With respect to **dependent claim 13**, Wen teaches the method wherein the changing icon appearance of the display system comprises:

determining whether the inputs for the new icon appearance are supported by the display system (See column 4, lines 30-61 and column 3, lines 1-22) and changing at least one of an icon size, vertical icon spacing, horizontal icon spacing, icon font size, and icon font type of the display system according to the new icon appearance if the user inputs are supported by the display system(See Figure 2, the user can crop the image which effects its size, See also icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61).

.

Art Unit: 2179

With respect to **claims 14-22**, claims 14-22 reflect a system comprising computer readable instructions for performing the steps of method claims 1-9, respectively and in further view of the following are rejected along the same rationale. Wen teaches a system (See column 2, bottom and Figure 1).

With respect to **claim 23**, claim 23 reflects a computer product comprising computer readable instructions executed by a processor for performing the steps of method claim 1 and in further view of the following is rejected along the same rationale. Wen teaches the system can be implemented on a conventional windows system which is a product on a medium (See column 3, middle and column 2, bottom).

With respect to **dependent claim 37**, Wen teaches the method wherein the display properties include one of an icon size, a vertical icon spacing, a horizontal icon spacing, an icon font size and an icon font size (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61)..

With respect to **dependent claim 38**, Wen teaches the method wherein the change in the sample icon's appearance is performed with respect to the backed-up display properties (See column 4, lines 30-67).

With respect to **dependent claim 39**, Wen teaches the method further comprising, prior to the changing the icon appearance of the display system:

temporarily storing the display properties of the display system, which correspond to a current icon appearance, in a memory location different from where the display properties of the display system, which correspond to the original icon appearance, are backed-up (See column 4, lines 30-67).

.

With respect to **dependent claim 40**, Wen teaches the method further comprising in response to the user's first command, restoring the changed display properties to the temporarily stored display properties (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61).

Art Unit: 2179

in response to the user's second command different from the first command, restoring the changed display properties to the backed-up display properties (See Figure 12 and column 5, lines 60-67 and column 6, lines 1-29).

With respect to **dependent claim 41**, Wen teaches the method further comprising:

prior to the changing the icon appearance of the display system, temporarily storing the display properties of the display system which correspond to a current icon appearance (See column 4, lines 30-67). And , in response to the user's first command, restoring the changed display properties to the temporarily stored display properties(See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61) And, in response to the user's second command different from the first command, restoring the changed display properties to the backed-up display properties(See Figure 12 and column 5, lines 60-67 and column 6, lines 1-29).

With respect to **dependent claim 42**, as indicated in the above discussion Wen in view of Straub teaches every element of claim 10.

Wen does not expressly teach the method further comprising:

if the display properties are determined to be invalid, changing the invalid display properties to valid display properties before said generating the first registry sub key.

Straub teaches a dynamic user interface update system that allows the user to make changes to icons, display colors, backgrounds, etc and in response to the update a service is activated to backup the display properties in the registry 66, when a smart folder update is specified (see column 8, lines 25-45). Straub teaches a registry that stores display information in a database, which contains tables. The web update service is set when the user changes the interface display properties from a window or dialog box (See column 8, top). Straub teaches the web update service "can" perform updates by unattended

Art Unit: 2179

scheduled downloading and "can also" perform updates by background downloading when the system is connected to the Internet and performs the update "automatically" according to the registry entries (see col. 10, lines 1-31). Straub also verifies the "validity" of the WCF (web collection file) using digital signatures where the WCF points to the new interface theme icons. Thus Straub "suggests" automatically updating an icon, where the user makes adjustments to the displayed theme and the registry entry and web update are updated. The system verifies the validity of the theme change and icon and automatically backs up the display properties via the web update service. Straub and Wen teach updating a user desktop appearance, via dialog boxes open on the interface. They both teach updating registry files. They both teach allowing the user to specify how the appearance of the icon is set. The difference between Straub and Wen is in Wen the user specifies the path in the registry to store the old icon and in Straub the system automatically updates a folder that refers to the registry entries when the user makes an update. Nonetheless, the outcome on the interface for both will result in a change to an icon appearance by changing the registry values used by the interface to point to the new image location when rendering the interface.

Accordingly, it would have been obvious to the skilled artisan at the time of the invention having the teachings of Straub and Wen in front of them to modify the manual process of updating registry values with the automatic system of Straub for the purposes of updating the interface without user input and to keep track of changes to interface appearances (see column 6, lines 55-67, column 7, lines 1-15, and 50-67, column 8, top, column 10, lines 1-31 and column 14, lines 25-67 and column 15, lines 1-45). The motivation to combine Straub with Wen comes from the suggestion in Straub that the windows interface provides for many features that allow the user to perform interface customizations to icons and other interface elements (See column 2, lines 12-55 and col. 6, lines 55-67) and to store and automatically update the interface elements on the computer (See col. 7, lines 1-50).

With respect to **dependent claim 53**, Wen teaches the method wherein the change in the sample icon's appearance is performed without changing the icon appearance of the display system (See column 3,

Art Unit: 2179

lines 1-22) Wen teaches the user can select the done button and the system will not make the actual changes to the system, .

With respect to **dependent claim 54**, Wen teaches the method wherein the icon control window comprises:

an icon appearance controller that receives the user inputs for a new icon appearance (see Figure 2 and column 3, lines 1-22).

a preview region that includes the at least one sample icon, the appearance of the sample icon being automatically changed in response to the user inputs received through the icon appearance controller; and(see Figure 2 and column 3, lines 1-22 and Figure 7).

an execution controller that interfaces with the display system and receives the user inputs to change the icon appearance of the display system (See Fig. 1 and Fig. 7 and column 4, lines 30-62).

wherein the icon appearance of the display system is changed only in response to the user inputs received through the execution controller and not in response to the user inputs received through the icon appearance controller (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61). See interface manager vs. cropping manager.

6. **Claims 43-49 and 52 are rejected under 35 U.S.C 103(a) as being unpatentable over Wen et al. (hereinafter Wen) U.S. Patent No. 6590590 issued July 8, 2003, in view of Straub et al. (hereinafter Straub) U.S. Patent No. 6091411 issued July 18, 2000, in further view of Axialis IconWorkshop et al (hereinafter Axialis).**

In regard to **Independent claim 43**, Wen teaches a method of controlling an icon appearance of a display system having a display screen, the method comprising:

Art Unit: 2179

- Displaying an icon control window on the display screen, the icon control window including a sample icon for a user's preview (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61). Wen teaches a control window that allows the user to see an icon and a preview of the sampled icon in the preview window.
- Selecting one sample icon of the icon control window according to inputs for a new icon appearance being received from a user through the icon control window (See Figure 2, icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-21 and column 4, lines 33-61). Wen teaches the user can change the appearance of the icon by cropping it and the cropped image of the icon will be shown in the preview window. The user manipulates the cropping window to change the image.
- Changing the icon appearance of the display system by changing the display properties in accordance with the user inputs (See column 4, lines 33-61 and column 6, lines 8-12 and figure 7). Wen shows in figure 7 that the update to change the icon appearance is reflected in the control window and the appearance is changed when the user crops or adjusts the image. The update to the new icon is performed prior to viewing the sample icon in the preview window. Wen teaches a preview window that displays a preview icon of the user's manipulated icon in the crop interface and the crop icon can be considered a sample icon, the text does not expressly recite that the crop window 21 displays the sample icon. The text recites displaying the old icon, while allowing the user to adjust the old icon displaying the preview icon. The user can nonetheless make multiple crop changes to window 21 without actually changing the system icon appearance. The window 21 is considered as displaying at least one sample icon that is visible to the user. The user makes a change and a preview of the icon is displayed, adjacent to the adjusted icon.

Art Unit: 2179

Wen does not expressly teach:

- a plurality of sample icons having different sizes for a user's preview
- Selecting one sample icon among the sample icons of the icon control window according to inputs

Axialis teaches a Icon Workshop interface editing system that allows the user to see multiple preview or sample icons while making icon appearance changes (See Para 2, 3, 13-22, specifically page 21, that allows the user to see several sample icons while changing the size of the icon). Axialis also teaches an editor window view and it's built in explorer to browse icons and see the icons in a preview thumbnail (See page 50, middle top). Axialis teaches a editor that allows the user to make specific display appearance changes to an icon while displaying a preview or sample icon (See page 54) and a process to changes system icons (See page 57 and 114 - 116) by using a customize windows icon dialog box similar to Wen. Straub, Wen and Axialis are analogous art because they all teach a process of changing a displayed icons appearance. They all perform the function via a dialog with the user and they all allow the user to alter the appearance of the interface icon.

Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the invention having the teachings of Wen and Axialis in front of them to modify the system of Wen to show sample icons to the user comprising icons of different display properties such as size and color, etc. The motivation to combine Axialis with Wen comes from within Axialis to create different size icons so that the icons will be compatible with different screen configurations and so that the icon image formats will be valid with the current windows icon formats (See page 11 and 12). Moreover for the expressed feature of allowing the user to customize their windows icons (See page 2).

Art Unit: 2179

With respect to **dependent claims 44 -49**, as indicated in the above discussion, Wen in view of Axialis teaches each element of claim 1.

Wen does not expressly teach:

- wherein exactly one of the sample icons of the icon control window has a size identical to the current icon size of the display system.
- an icon size controller providing a plurality of selectable icon sizes for the user to select a desired icon size from the selectable icon sizes;
- a preview region including the at least one sample icon, the sample icon being resized when the desired icon size is selected through the icon size controller;
- an execution controller interfacing with the display system in order to change an icon size of the display system according to the selected icon size.
- wherein the icon size controller comprises a sliding bar with minimum and maximum icon sizes, the user selecting the desired icon size by moving a size indicator within the sliding bar.
- wherein the minimum and maximum icon sizes of the sliding bar are selected from a size range supported by the display system.
- wherein the icon size controller comprises a plurality of selectable buttons representing the plurality of selectable icon sizes, the user selecting the desired icon size by selecting one of the selectable buttons.
- wherein the plurality of selectable buttons include toggle buttons.

Axialis teaches a Icon Workshop interface editing system that allows the user to see multiple preview or sample icons while making icon appearance changes (See Para 2, 3, 13-22, specifically page 21, that allows the user to see several sample icons while changing the size of the icon). Axialis also teaches an editor window view and it's built in explorer to browse icons and see the icons in a preview thumbnail (See page 50, middle top). Axialis shows a plurality of size of the icons displayed in the interface and a sliding bar that allows the user to scroll through the icon sizes. Axialis teaches a table that is shown to the user to indicate the max. and min icon size for the given operating system. The user can browse the icon images and select or unselect the sample icons they wish to see within the interface. Axialis teaches a editor that allows the user to make specific display appearance changes to an icon while displaying a preview or sample icon (See page 54) and a process to changes system icons (See page 57 and 114 -

Art Unit: 2179

116) by using a customize windows icon dialog box similar to Wen. Straub, Wen and Axialis are analogous art because they all teach a process of changing a displayed icons appearance. They all perform the function via a dialog with the user and they all allow the user to alter the appearance of the interface icon.

Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the invention having the teachings of Wen and Axialis in front of them to modify the system of Wen to show sample icons to the user comprising icons of different display properties such as size and color, etc. The motivation to combine Axialis with Wen comes from within Axialis to create different size icons so that the icons will be compatible with different screen configurations and so that the icon image formats will be valid with the current windows icon formats (See page 11 and 12). Moreover for the expressed feature of allowing the user to customize their windows icons (See page 2).

With respect to **dependent claim 52**, Wen teaches the method wherein the changing icon appearance of the display system comprises:

determining whether the inputs for the new icon appearance are supported by the display system (See column 4, lines 30-61 and column 3, lines 1-22) and changing at least one of an icon size, vertical icon spacing, horizontal icon spacing, icon font size, and icon font type of the display system according to the new icon appearance if the user inputs are supported by the display system (See Figure 2, the user can crop the image which effects its size, See also icon control window 21, and preview 22, and icon update process figure 3, and Figure 7 and column 2, bottom and column 3, lines 1-40 and column 4, lines 33-61).

7. **Claims 50-51 are rejected under 35 U.S.C 103(a) as being unpatentable over Wen et al. (hereinafter Wen) U.S. Patent No. 6590590 issued July 8, 2003, in view of Straub et al.**

Art Unit: 2179

(hereinafter Straub) U.S. Patent No. 6091411 issued July 18, 2000, in view of Axialis

IconWorkshop et al (hereinafter Axialis), in further view of Straub

With respect to **dependent claims 50-51**, as indicated in the above discussion, Wen in view of Axialis teaches every element of claim 43.

Wen in view of Axialis do not expressly recite where the display properties are determined to be valid based on a display properties table of the display system and displaying the icon control window on the display screen if the display properties are determined to be valid.

Straub teaches a dynamic user interface update system that allows the user to make changes to icons, display colors, backgrounds, etc and in response to the update a service is activated to backup the display properties in the registry 66, when a smart folder update is specified (see column 8, lines 25-45). Straub teaches a registry that stores display information in a database, which contains tables. The web update service is set when the user changes the interface display properties from a window or dialog box (See column 8, top). Straub teaches the web update service "can" perform updates by unattended scheduled downloading and "can also" perform updates by background downloading when the system is connected to the Internet and performs the update "automatically" according to the registry entries (see col. 10, lines 1-31). Straub also verifies the "validity" of the WCF (web collection file) using digital signatures where the WCF points to the new interface theme icons. Thus Straub "suggests" automatically updating an icon, where the user makes adjustments to the displayed theme and the registry entry and web update are updated. The system verifies the validity of the theme change and icon and automatically backs up the display properties via the web update service. Straub and Wen teach updating a user desktop appearance, via dialog boxes open on the interface. They both teach updating registry files. They both teach allowing the user to specify how the appearance of the icon is set. The difference between Straub and Wen is in Wen the user specifies the path in the registry to store the old icon and in Straub the system automatically updates a folder that refers to the registry entries when the user makes an update. Nonetheless, the outcome on the interface for both will result in a change to an icon appearance by changing the registry values used by the interface to point to the new image location when rendering the interface.

Art Unit: 2179

Accordingly, it would have been obvious to the skilled artisan at the time of the invention having the teachings of Straub and Wen in front of them to modify the manual process of updating registry values with the automatic system of Straub for the purposes of updating the interface without user input and to keep track of changes to interface appearances (see column 6, lines 55-67, column 7, lines 1-15, and 50-67, column 8, top, column 10, lines 1-31 and column 14, lines 25-67 and column 15, lines 1-45). The motivation to combine Straub with Wen and Axialis comes from the suggestion in Straub that the windows interface provides for many features that allow the user to perform interface customizations to icons and other interface elements (See column 2, lines 12-55 and col. 6, lines 55-67) and to store and automatically update the interface elements on the computer (See col. 7, lines 1-50).

A reference to specific paragraphs, columns, pages, or figures in a cited prior art reference is not limited to preferred embodiments or any specific examples. It is well settled that a prior art reference, in its entirety, must be considered for all that it expressly teaches and fairly suggests to one having ordinary skill in the art. Stated differently, a prior art disclosure reading on a limitation of Applicant's claim cannot be ignored on the ground that other embodiments disclosed were instead cited. Therefore, the Examiner's citation to a specific portion of a single prior art reference is not intended to exclusively dictate, but rather, to demonstrate an exemplary disclosure commensurate with the specific limitations being addressed. In re Heck, 699 F.2d 1331, 1332-33, 216 USPQ 1038, 1039 (Fed. Cir. 1983) (quoting In re Lemelson, 397 F.2d 1006, 1009, 158 USPQ 275, 277 (CCPA 1968)). In re: Upsher-Smith Labs. v. Pamlab, LLC, 412 F.3d 1319, 1323, 75 USPQ2d 1213, 1215 (Fed. Cir. 2005); In re Fritch, 972 F.2d 1260, 1264, 23 USPQ2d 1780, 1782 (Fed. Cir. 1992); Merck & Co. v. Biocraft Labs., Inc., 874 F.2d 804, 807, 10 USPQ2d 1843, 1846 (Fed. Cir. 1989); In re Fracalossi, 681 F.2d 792, 794 n.1, 215 USPQ 569, 570 n.1 (CCPA 1982); In re Lamberti, 545 F.2d 747, 750, 192 USPQ 278, 280 (CCPA 1976); In re Bozek, 416 F.2d 1385, 1390, 163 USPQ 545, 549 (CCPA 1969).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to STEVEN B. THERIAULT whose telephone number is (571)272-5867. The examiner can normally be reached on Mon.-Fri. 10 am - 7 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Weilun Lo can be reached on (571) 272-4847. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2179

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Steven B Theriault/
Primary Examiner
Art Unit 2179